Advances in Information Mining
ISSN: 0975-3265 & E-ISSN: 0975-9093, Volume 4, Issue 1, 2012, pp.-57-66.
Available online at http://www.bioinfo.in/contents.php?id=32

PERFORMANCE COMPARISON OF TIME SERIES DATA USING PREDICTIVE DATA MINING TECHNIQUES

SAIGAL S.[1]* AND MEHROTRA D.[2]
[1]Department of Computer Science, Krishna Engineering College, Ghaziabad - 201007, UP, India.
[2]Department of MCA, Amity School of Computer Sciences, Noida-201303, UP, India.
*Corresponding Author: Email- saigalsheetal9@gmail.com

Received: November 09, 2012; Accepted: December 03, 2012

Abstract- This paper focuses on the methodology used in applying the Time Series Data Mining techniques to financial time series data for calculating currency exchange rates of US dollars to Indian Rupees. Four Models namely Multiple Regression in Excel, Multiple Linear Regression of Dedicated Time Series Analysis in Weka, Vector Autoregressive Model in R and Neural Network Model using NeuralWorks Predict are analyzed. All the models are compared on the basis of the forecasting errors generated by them. Mean Error (ME), Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Square Error (RMSE), Mean Percentage Error (MPE) and Mean Absolute Percentage Error (MAPE) are used as a forecast accuracy measure. Results show that all the models accurately predict the exchange rates, but Multiple Linear Regression of Dedicated Time Series Analysis in Weka outperforms the other three models.
Keywords- Exchange Rate Prediction, Time Series Models, Regression, Predictive Data Mining, Weka, VAR, Neural Network

Citation: Saigal S. and Mehrotra D. (2012) Performance Comparison of Time Series Data Using Predictive Data Mining Techniques. Advances in Information Mining, ISSN: 0975-3265 & E-ISSN: 0975-9093, Volume 4, Issue 1, pp.-57-66.

Copyright: Copyright©2012 Saigal S. and Mehrotra D. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution and reproduction in any medium, provided the original author and source are credited.

Introduction

One of the most enticing application areas of data mining in these emerging technologies is in finance, becoming more amenable to data-driven modeling as large sets of financial data become available. In the field of finance the extensive use of data mining applications includes the area of forecasting stock market, pricing of corporate bonds, understanding and managing financial risk, trading futures, prediction of exchange rates, credit rating etc.

Monthly data is collected for the last 10 years from 2000 to 2010, for predicting exchange rates of 2011 [5,14,16].The original rate of 2011 is available and then compared with the predicted values for calculating the accuracy of the models. Mean Error (ME), Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Square Error (RMSE), Mean Percentage Error (MPE) and Mean Absolute Percentage Error (MAPE) is used as a forecast accuracy measure. The multiple variables used on which exchange rate depends are CPI, Trade Balance (in million US dollars), GDP, Unemployment and Monetary Base (in billion dollars) [16].

Four Models namely Multiple Linear Regression in Excel [14], Multiple Linear Regression of Dedicated Time Series Analysis in Weka [6, 9], Vector Autoregressive Model in R [6-8, 10] and Neural Network Model [4,5,13-15] using NeuralWorks Predict are analyzed. All the models are compared on the basis of the forecasting errors generated by them.

The paper is organized as follows. Section II covers predictive data mining. Section III covers the four predictive time series models namely Multiple Linear Regression in Excel, Multiple Linear Regression of Dedicated Time Series Analysis in Weka, Vector Autoregressive Model in R and Neural Network Model using NeuralWorks Predict. Section IV covers the datasets used for the analysis and the steps and results obtained by using the four models. Section V shows the performance comparison of the four models using Mean Error (ME), Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Square Error (RMSE), Mean Percentage Error (MPE) and Mean Absolute Percentage Error (MAPE). Section VI concludes the work, followed by references in the last section.

Predictive Data Mining

Predictive data mining analyzes data in order to construct one or a set of models and attempts to predict the behavior of new data sets. Prediction is a form of data analysis that can be used to extract models describing important data classes or to predict future data trends. Such analysis can help provide us with a better understanding of the data at large.

Prediction can also be viewed as a mapping or function, y = f (X), where X is the input (e.g., a tuple describing a loan applicant), and the output *y* is a continuous or ordered value (such as the predicted amount that the bank can safely loan the applicant); That is, we wish to learn a mapping or function that models the relationship between X and *y*.

There are two issues regarding prediction: first is preparing the data for prediction which involves the preprocessing steps like data cleaning, relevance analysis, data transformation and data reduction, second issue is comparing the different prediction models. The models are compared according to the criteria given below:

- **Accuracy:** The accuracy of a predictor refers to how well a given predictor can guess the value of the predicted attribute for new or previously unseen data.
- **Speed:** This refers to the computational costs involved in generating and using the given predictor.
- **Robustness:** This is the ability of the predictor to make correct predictions given noisy data or data with missing values.
- **Scalability:** This refers to the ability to construct the predictor efficiently given large amounts of data.
- **Interpretability:** This refers to the level of understanding and insight that is provided by the predictor.

Although accuracy, speed, robustness, scalability and interpretability are the various factors for comparing the prediction models, but in this paper the prediction models are compared on the basis of their accuracy. Measures like ME, MAE, MSE, RMSE, MPE and MAPE are used to compare the performance of predictive models [18].

Predictive data-mining tools are designed to help us understand what the useful information looks like and what has happened during past procedures. Therefore, the tools use the description of the useful information to find similar examples of hidden information in the database and use the information learned from the past to develop a predictive model of what will happen in the future. Different predictive models are analyzed and the best model is chosen for predicting the data. [Fig-1] below shows that the best model is chosen among different sized models to get required solution.

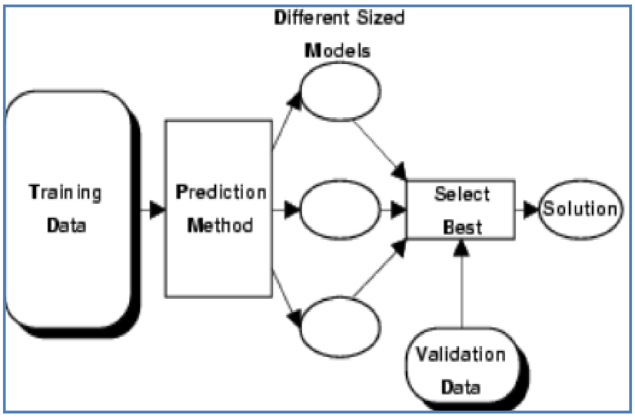

Fig. 1- Find the best model among predictive model[17]

Predictive Time Series Models

A Time Series is a time-oriented or chronological sequence of observations on a variable of interest. Time series analysis is the process of using statistical techniques to model and explain a time-dependent series of data points. Time series forecasting is the process of using a model to generate predictions (forecasts) for future events based on known past events. Time series data has a natural temporal ordering - this differs from typical data mining/ machine learning applications where each data point is an independent example of the concept to be learned, and the ordering of data points within a data set does not matter. Time series prediction proposes algorithms for which past data (mainly finite observation sequences of data points related to uniform time interval) are used to generate models to forecast future data points of the series [1-3,11,12].

There are only two broad types of forecasting techniques- Qualitative methods and Quantitative methods. Qualitative forecasting techniques are often subjective in nature and require judgment on the part of experts. Quantitative forecasting techniques make formal use of historical data and a forecasting model.

In this analysis, four models namely Multiple Linear Regression in Excel, Multiple Linear Regression of Dedicated Time Series Analysis in Weka, Vector Autoregressive Model in R and Neural Network Model using NeuralWorks Predict is analyzed.

Multiple Linear Regression in Excel

Regression models make use of relationships between the variable of interest and one or more related predictor variables. Sometimes regression models are called causal forecasting models, because the predictor variables are assumed to describe the forces that cause or drive the observed values of the variable of interest.

Regression analysis is a statistical technique for modeling and investigating the relationships between an outcome or response variable and one or more predictor or regressor variables. The end result of a regression analysis study is often to generate a model that can be used to forecast or predict future values of the response variable given specified values of the predictor variables.

The Multiple Linear Regression Model is

$$y = \beta_0 + \beta_1 x + \beta_1 x^2 + \in \quad (1)$$

The parameters β_0, $\beta_1 \ldots$ β_2 in this model are often called partial regression coefficients because they convey information about the effect on *y* of the predictor that they multiply given that all of the other predictors in the model do not change. The regression models in [Eq-1] is linear regression models because they are linear in the unknown parameters (the β's), and not because they necessarily describe linear relationships between the response and the regressors.

Multiple Linear Regression of Dedicated Time Series Analysis in Weka

WEKA, formally called Waikato Environment for Knowledge Learning, is a computer program that was developed at the University of Waikato in New Zealand for the purpose of identifying information from raw data gathered from agricultural domains. WEKA supports many different standard data mining tasks such as data preprocessing, classification, clustering, regression, visualization and feature selection. Weka (>= 3.7.3) now has a dedicated time series analysis environment that allows forecasting models to be developed, evaluated and visualized. This environment takes the form of a plugin tab in Weka's graphical "Explorer" user interface and can be installed via the package manager. Weka's time series framework takes a machine learning/data mining approach to modeling time series by transforming the data into a form that standard propositional learning algorithms can process. It does this by removing the temporal ordering of individual input examples by encoding the time dependency via additional input fields. These fields are sometimes referred to as "lagged" variables. Various other fields are also computed automatically to allow the algorithms to model trends and seasonality. After the data has been transformed, any of Weka's regression algorithms can be applied to learn a model. An obvious choice is to apply multiple linear regression, but any method capable of predicting a continuous target can be applied - including powerful non-linear methods such as support vector machines for regression and model trees. This approach to time series analysis and forecasting is often more powerful and more flexible that classical statistical techniques such as ARMA and ARIMA.

Vector Autoregressive Model in R

VAR analysis has evolved as a standard instrument in econometrics for analyzing multivariate time series. A VAR consists of a set of K endogenous variables $y_t = (y_{1t}, y_{kt}, \ldots\ldots y_{Kt})$ for k = 1,.... K. The VAR (p)-process is then defined as:

$$y_t = A_1 y_{t-1} + \ldots + A_p y_{t-p} + u_t \qquad (2)$$

with A_i are (K × K) coefficient matrices for i = 1,..., p and u_t is a K-dimensional white noise process with time invariant positive definite covariance matrix E ($u_t u_t'$) = $\sum u$. [Eq-2] is sometimes written in the form of a lag polynomial

$$A(L) = (lK - A_1 - \ldots - A_p) \ as \ A(L)\, y_t = CD_t + u_t \qquad (3)$$

where the matrix C is the coefficient matrix of potentially deterministic regressors with dimension (K ×M), and D_t is an (M ×1) column vector holding the appropriate deterministic regressors, such as a constant, trend, and dummy and/or seasonal dummy variables

One important characteristic of a VAR (p)-process is its stability. The stability of an empirical VAR(p)-process can be analyzed by considering the companion form and calculating the eigenvalues of the coefficient matrix. If the moduli of the eigenvalues of A are less than one, then the VAR (p)-process is stable.

The information criteria are implemented in the functions VAR () and VARselect () contained in the package vars. In the former function, an appropriate VAR (p)-model will be estimated by providing the maximal lag number, lag.max, and the desired criterion. The calculations are based upon the same sample size. That is, lag.max values are used as starting values for each of the estimated models. The result of the function VARselect () is a list object with elements selection and criteria. The element selection is a vector of optimal lag length according to the above-mentioned information criteria. The element criteria is a matrix containing the particular values for each of these criteria up to the maximal lag order chosen.

Neural Network Model using NeuralWorks Predict

Artificial neural networks were inspired in large part by research into the function of neurons in the human brain. Artificial neural networks process information in a way that resembles the way the brain works. Like the brain, neural networks “learn” from experience during a process called training.

We can use neural networks when unknown relationships exist in historical data. A historical dataset consists of recorded input values and their corresponding output values. Neural networks can detect patterns in data, generalize about relationships in data, and generate an output value when given a new set of input values from the problem domain. Analysts who lack extensive domain knowledge can use neural networks to solve problems that prove too complex for more conventional analytical techniques.

How a Neural Network Learns

The human brain is a very complex system of interconnected neurons. Similarly, a neural network is an interconnected system of artificial “neurons.” In neural network terminology, neurons are called Processing Elements or nodes. Like a neuron in the brain, each Processing Element (PE) can accept input data, process the data, and pass it to the next PE. A PE processes data using one of several types of mathematical functions. In effect an entire neural network represents a composite of the functions represented by all PEs.

The key to building a robust neural network is to collect many examples (or records) of input values and corresponding output values over time. The neural network uses this historical data to determine (learn) a mathematical relationship between the input data and the output data.

Network Architecture

In a neural network, PEs can be interconnected in various ways. Typically, PEs are structured into layers and the output values of PEs in one layer serve as input values for PEs in the next layer. Each connection has a weight associated with it. In most cases, a Processing Element calculates a weighted sum of incoming values (the sum the outputs of the PEs connected from the next lower level multiplied by their connection weights). This sum is called the activation value. The activation value is then passed to the PE’s non-linear transfer function to produce an output for that PE. This combination of PEs, connections, weights, and transfer functions form the network architecture. This architecture then represents a complex mathematical formula that has been derived from historical data.

Training a Neural Network

Neural networks learn from “experience” (exposure to information). From repeated exposure to historical data, a neural network learns to strengthen connection weights from PEs that have a greater tendency to accurately predict the desired output. The strength of each connection (the magnitude of the connection weight) increases or decreases based on its influence in producing the output associated with each input data record in the historical dataset. Connection weights are adjusted during training.

Training is the process of repeatedly (iteratively) exposing a neural network to examples of historical data. Each example contains input variables, and the associated desired output. The desired output is the value that the neural network should predict, given the associated input values in the historical data. In neural network terminology, the desired output is called the target value. During training, network weights are adjusted according to a learning rule- an algorithm that specifies how the error between predicted outputs and target values should be used to modify network weights. Before training begins, a simple network with the necessary number of input and output PEs is created. All connection weights are initialized to small random values. Data examples from the training set are passed to the network and the network produces an output value. This value is compared to the target value. The weights are adjusted in order to decrease the error between the network output and the target value. In addition, more processing elements are added to the network if doing so helps decrease the error of the network. Training continues until the neural network produces output values that match the target values within a specified accuracy level, or until some other stopping criterion is satisfied.

Testing and Validating a Neural Network

Just as you might test a person’s skill in a controlled environment, you test a neural network using historical data it has not seen. Test results are good if the predicted values are close to the target values.

One difficulty for neural networks (and other non-linear estimation techniques) is the possibility that the network will over-fit the training data. This means that the network might closely predict the target values on the training data but produce inferior results for new data. Training for too long (too many passes through the training set) can cause the function to become very complex in order to produce the target values at the expense of generalizing well on unseen data. However, if a network is not trained long enough, it doesn't fully learn trends and relationships in the data.

One way of knowing when to stop training is to automatically and periodically test the performance on a test set during training. When the performance starts to degrade on the test set, it's time to stop training-the neural network has started to learn relationships which are specific to only the training set! At the end of training, the neural network can be further tested on an additional independent test set referred to as a validation set.

Using a Neural Network

After the neural network has been trained, it can be used to make predictions given input data for which the desired output value is unknown.

NeuralWorks Predict

NeuralWorks Predict is a complete application development environment for creating and deploying real-time applications for forecasting, modeling, classification and clustering or grouping. This powerful system combines neural network technology with fuzzy logic, statistics and genetic algorithms to find solutions to forecasting, modeling and classification problems automatically. Key features of NeuralWorks Predict are:

- **NeuralWorks Predict Understands Data:** NeuralWorks Predict knows how to transform data into forms that maximize the power of neural network techniques. This saves time and also results in better performance.
- **NeuralWorks Predict Applies the Rigors of Statistics to Model the Real World:** The algorithms used in NeuralWorks Predict are grounded in statistics. Key concepts such as ridge regression, maximum likelihood and cross-validation are seamlessly integrated with neural techniques. NeuralWorks Predict models are statistically based neural models.
- **Simple is Better:** NeuralWorks Predict tackles complex problems, identifies the most salient features and builds just the right size model to solve your problem. NeuralWorks Predict has been used to sort through thousands of variables. Simple models perform better on new data, are faster and more reliable than more complex models built by trial and error.
- **NeuralWorks Predict Models are a Hybrid of Polynomial Regression, Fuzzy Logic and Neural Networks:** This includes polynomial neural and fuzzy-neural regression, data distribution compensation, non-parametric outlier detection and transformation, stochastic gradient search and mixed-mode hidden-layer functions (sigmoid, tanh, sine, exponential, linear). NeuralWorks Predict picks the right building blocks to build the best solution.
- **NeuralWorks Predict Handles the Big Problems:** The command line version of NeuralWorks Predict eliminates the barriers on data set size allowing 4,000 fields. The number of records is only limited by the system's memory.

Empirical Results

Datasets

Time Series Data Mining methods are applied to financial time series for calculating currency exchange rates of US dollars to Indian Rupees. Monthly data is collected for the last 10 years from 2000 to 2010, for predicting exchange rates of 2011. The original rate of 2011 is available and then compared with the predicted values for calculating the accuracy of the models. The multiple variables used on which exchange rate depends are CPI, Trade Balance (in million US dollars), GDP, Unemployment and Monetary Base (in billion dollars). The data for all the variables is collected from various financial sites like tradingeconomics, ycharts, census.gov etc. The explanation of the variables is:

- **CPI:** Consumer Price Index (CPI) data measures inflation in an economy. When inflation gets too high in a country, the Central bank may increase interest rates in order to ensure price stability. This may cause the currency to rise in value as the additional interest received makes the currency more desirable.
- **Trade Balance:** A country's balance of trade is the total value of its exports, minus the total value of its imports. If this number is positive, the country is said to have a favorable balance of trade. If the difference is negative, the country has a trade gap, or trade *deficit.*
- **GDP:** Gross Domestic Product is a measure of the overall economic output within a country's borders over a particular time, typically a year. GDP is calculated by adding together the total value of annual output of all that country's goods and services. GDP can also be measured by income by considering the factors producing the output-the capital and labor-or by expenditure by government, individuals, and business on that output
- **Unemployment:** Employment levels have an immediate impact on economic growth. An increase in unemployment signals a slowdown in the economy and possible devaluation of a country's currency because of declining confidence and lower demand.
- **Monetary Base:** In economics, the monetary base (also base money, money base, high-powered money, reserve money, or, in the UK, narrow money) is a term relating to (but not being equivalent to) the money supply (or money stock), the amount of money in the economy. The monetary base is highly liquid money that consists of coins, paper money (both as bank vault cash and as currency circulating in the public), and commercial banks' reserves with the central bank.

Steps and Results of Time Series Data

The steps of the models and the corresponding results obtained by using the four models are:

Multiple Linear Regression in Excel

- Steps for performing this Model are:
- Load the data in excel file.
- Generate the regression equation by applying the Multiple Linear Regression.
- Forecast the data for the specified year.
- Compare the forecasted value with the actual value.

- Compute ME, MAE, MSE, RMSE, MPE and MAPE.

[Fig-2] represents graph showing comparison between actual exchange rate of 2011 (shown by blue line) and predicted exchange rate of 2011 (shown by red line) using Multiple Linear Regression in Excel. The x-axis of the graph shows time period in months and the y-axis of the graph shows exchange rate values.

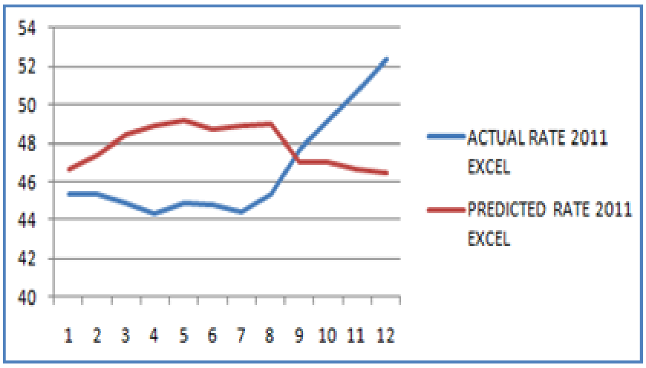

Fig. 2- Graph showing comparison between actual exchange rate of 2011 with predicted exchange rate of 2011 using Excel

The monthwise actual exchange rate of 2011 is known to us. [Table-1] shows the actual and predicted exchange rate of 2011 monthwise using Multiple Linear Regression in Excel.

Table 1- Compares the actual exchange rate of 2011 with predicted exchange rate of 2011

Time Period	Actual Exchange Rate	Predicted Exchange Rate
Jan-11	45.375	46.693815
Feb-11	45.3795	47.398572
Mar-11	44.9143	48.445847
Apr-11	44.301	48.865287
May-11	44.9024	49.142934
Jun-11	44.8109	48.749785
Jul-11	44.396	48.935335
Aug-11	45.3135	49.0162
Sep-11	47.6905	47.039847
Oct-11	49.202	47.004323
Nov-11	50.6785	46.630227
Dec-11	52.3824	46.461066

The regression equation obtained by applying Multiple Linear Regression in Excel is:

Exchange Rate = 47.21221439+0.172232073 *CPI - 0.000266695 *Trade Balance - 3.118399431 *GDP - 0.023711504 *Unemployment + 0.003750279 *MonetoryBase

Multiple Linear Regression of Dedicated Time Series Analysis in Weka

Steps for performing this Model are:

- Load the data into Weka's Explorer.
- Select Advanced Configuration panel. It gives the user full control over a number of aspects of the forecasting analysis. These include the choice of underlying model and parameters, creation of lagged variables, creation of variables derived from a date time stamp, specification of "overlay" data, evaluation options and control over what output is created.
- Select Base learner panel. It provides control over which Weka learning algorithm and is used to model the time series. It also allows the user to configure parameters specific to the learning algorithm. Select linear support vector machine for Regression (Weka's SMOreg).
- Select the Lag creation panel. It allows the user to control and manipulate how lagged variables are created. Lagged variables are the main mechanism by which the relationship between past and current values of a series can be captured by propositional learning algorithms. For monthly data lags upto 12 times ahead are created.
- In Lag Creation panel, select Adjust for variance check box which allows the user to opt to have the system compensate for variance in the data. It does this by taking the log of each target before creating lagged variables and building the model. This can be useful if the variance (how much the data jumps around) increases or decreases over the course of time. Adjusting for variance may, or may not, improve performance.
- Select the Evaluation panel, it allows the user to select which evaluation metrics they wish to see, and configure whether to evaluate using the training data and/or a set of data held out from the end of the training data. The available metrics are Mean absolute error (MAE), Mean squared error (MSE), Root mean squared error (RMSE), Mean absolute percentage error (MAPE), Direction accuracy (DAC), Relative absolute error (RAE), Relative absolute error (RAE) and Root relative squared error (RRSE). Select checkboxes of required metrics.
- Select the Output panel, it provides options that control what textual and graphical output are produced by the system. Selecting Output predictions at step causes the system to output the actual and predicted values for a single target at a single step. Selecting Output future predictions beyond the end of series will cause the system to output the training data and predicted values (up to the maximum number of time units) beyond the end of the data for all targets predicted by the forecaster. The predictions at a specific step can be graphed by selecting the Graph predictions at step check box. Selecting the Graph target at steps checkbox allows a single target to be graphed at more than one step - e.g. a graph can be generated that shows 1-step-ahead, 2-step-ahead and 5-step ahead predictions for the same target. The selected results will be displayed by running the Model.
- Compare the forecasted value with the actual value. Compute ME, MAE, MSE, RMSE, MPE and MAPE.

[Fig-3] represents graph showing comparison between actual exchange rate of 2011 (shown by blue line) and predicted exchange rate of 2011 (shown by red line) using Multiple Linear Regression of Dedicated Time Series Analysis in Weka. The x-axis of the graph shows time period in months and the y-axis of the graph shows exchange rate values.

The monthwise actual exchange rate of 2011 is known to us. [Table-2] shows the actual and predicted exchange rate of 2011 monthwise using Multiple Linear Regression of Dedicated Time Series Analysis in Weka

[Fig-4], [Fig-5] and [Fig-6] shows the graphs showing exchange rates of 2000 to 2011, predictions at steps and predictions at targets respectively. These graphs are generated by Dedicated Time Series Analysis in Weka.

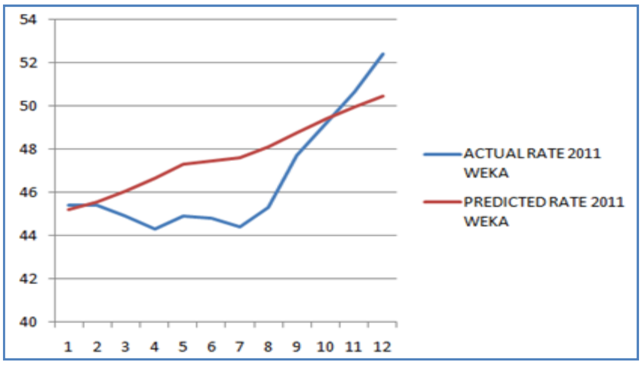

Fig. 3- Graph showing comparison between actual exchange rate of 2011 with predicted exchange rate of 2011 using Weka

Table 2- Compares the actual exchange rate of 2011 with predicted exchange rate of 2011

Time Period	Actual Exchange Rate	Predicted Exchange Rate
Jan-11	45.375	45.1854
Feb-11	45.3795	45.5341
Mar-11	44.9143	46.0724
Apr-11	44.301	46.6285
May-11	44.9024	47.2906
Jun-11	44.8109	47.4405
Jul-11	44.396	47.6117
Aug-11	45.3135	48.101
Sep-11	47.6905	48.776
Oct-11	49.202	49.395
Nov-11	50.6785	49.9483
Dec-11	52.3824	50.4683

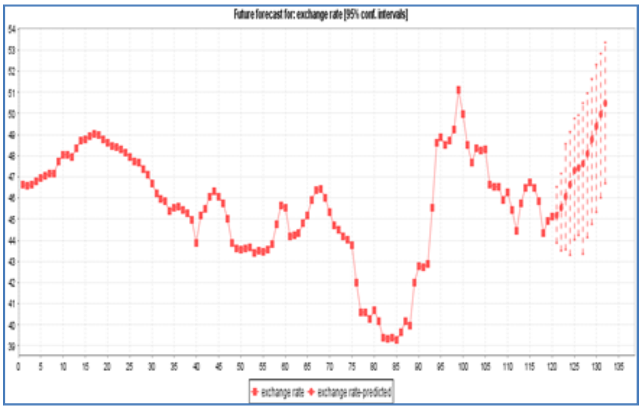

Fig. 4- Graph showing exchange rate for years 2000 to 2011

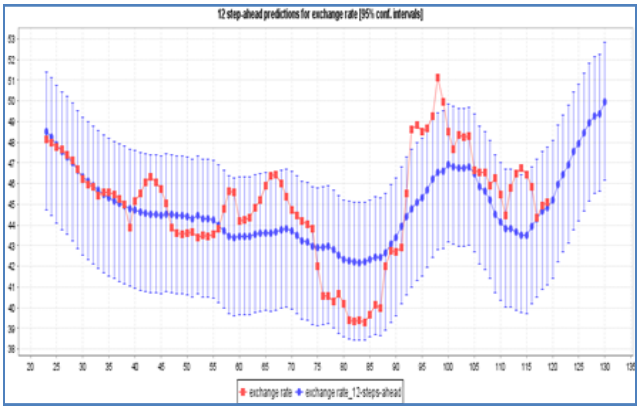

Fig. 5- Weka graph prediction at step

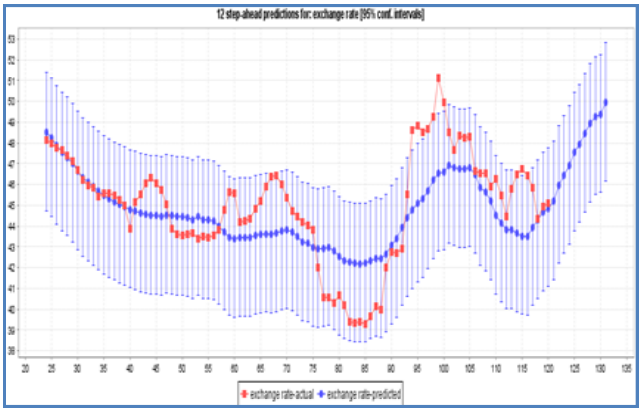

Fig. 6- Weka graph prediction at targets

[Fig-7] shows evaluation results using mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE) and mean squared error (MSE) for each step ahead in Weka.

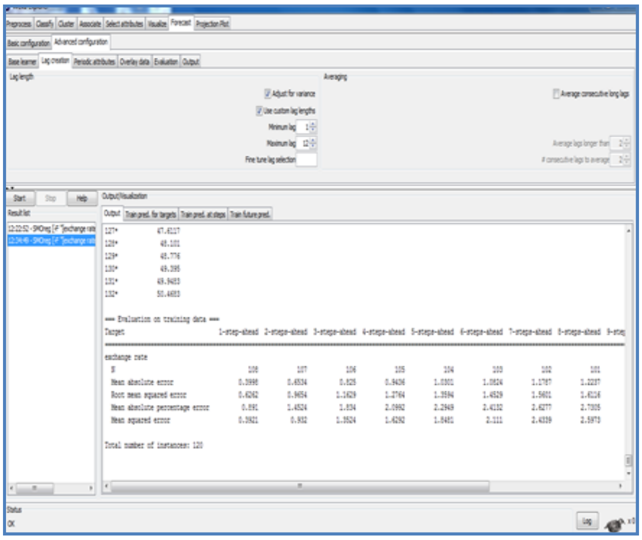

Fig. 7- Weka screenshot of evaluation results using various performance measures.

Vector Autoregressive Model in R

Steps for performing this Model are:

- Load the vars package.
- Load the data available in csv format.
- Plot the time series data of each variable.
- An optimal lag-order can be determined according to an information criteria or the final prediction error of a VAR (p) with the function VARselect (). This function returns a list object with the optimal lag-order according to each of the criteria, as well as a matrix containing the values for all lags up to lag.max.
- In a next step, the VAR (lag.max) is estimated with the function VAR () and as deterministic regressors a constant is included.
- Names () function returns a list object of class varest. Summary () method gives the regression results for various variables equations. A plot consisting of a diagram of fit, a residual plot, the autocorrelation and partial autocorrelation function of the residuals.

- We should check stability of VAR (p) process. Here, stability does not refer to the coefficients' stability, i.e. the stability of the regressions, but rather the stability of the system of difference equations. As pointed out above, if the moduli of the eigenvalues of the companion matrix are less than one, the system is stable. The moduli of the eigenvalues are found using roots () function.
- A predict-method for objects with class attribute varest is available for forecasting. The n.ahead forecasts are computed recursively for the estimated VAR, beginning with h = 1, 2,..., n. ahead. The default value of forecast confidence interval is 0.95. The predict-method does return a list object of class varprd with three elements. The first element, fcst, is a list of matrices containing the predicted values, the lower and upper bounds according to the chosen confidence interval, ci and its size. The second element, endog is a matrix object containing the endogenous variables and the third is the submitted varest object.
- A plot-method for objects of class varprd does exist as well as a fanchart() function for plotting fan charts. The fan chart () function has colors and cis arguments, allowing the user to input vectors of colors and critical values. If these arguments are left NULL, then as defaults a heat.map color scheme is used and the critical values are set from 0.1 to 0.9 with a step size of 0.1.
- Compare the forecasted value with the actual value. Compute ME, MAE, MSE, RMSE, MPE and MAPE.

[Fig-8] represents graph showing comparison between actual exchange rate of 2011 (shown by blue line) and predicted exchange rate of 2011 (shown by red line) using Vector Autoregressive Model in R. The x-axis of the graph shows time period in months and the y-axis of the graph shows exchange rate values.

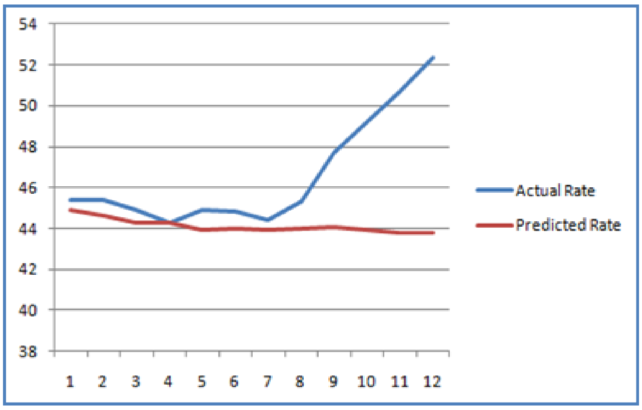

Fig. 8- Graph showing comparison between actual exchange rate of 2011 with predicted exchange rate of 2011 using R

[Table-3] shows fcst which is the list of predicted exchange rates of 2011, the lower and upper ranges for the result and the confidence interval.
The monthwise actual exchange rate of 2011 is known to us. [Table-4] shows the actual and predicted exchange rate of 2011 monthwise using Vector Autoregressive Model in R.
[Fig-9] and [Fig-10] shows the graphical representation and fan-chart representation of forecast of predicted exchange rate of 2011 using Vector Autoregressive Model in R respectively.

Table 3- Predicted currency exchange rate 2011 using R

FCST	LOWER	UPPER	CI
44.91408	43.76325	46.06492	1.150836
44.59574	42.76573	46.42575	1.830007
44.25757	41.97348	46.54167	2.284093
44.28305	41.65696	46.90914	2.626088
43.91192	41.01431	46.80954	2.897611
43.96878	40.77892	47.15865	3.189865
43.92951	40.44813	47.41089	3.48138
43.97694	40.23632	47.71757	3.740627
44.09229	40.10983	48.07475	3.982462
43.95191	39.79851	48.10532	4.153402
43.80791	39.51926	48.09656	4.28865
43.77771	39.36423	48.19119	4.41348

Table 4- Compares the actual exchange rate of 2011 with predicted exchange rate of 2011

Time Period	Actual Exchange Rate	Predicted Exchange Rate
Jan-11	45.375	44.91408
Feb-11	45.3795	44.59574
Mar-11	44.9143	44.25757
Apr-11	44.301	44.28305
May-11	44.9024	43.91192
Jun-11	44.8109	43.96878
Jul-11	44.396	43.92951
Aug-11	45.3135	43.97694
Sep-11	47.6905	44.09229
Oct-11	49.202	43.95191
Nov-11	50.6785	43.80791
Dec-11	52.3824	43.77771

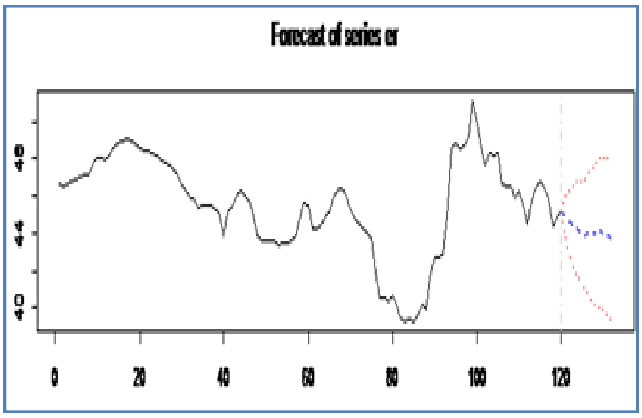

Fig. 9- Graph showing forecast of predicted exchange rates 2011

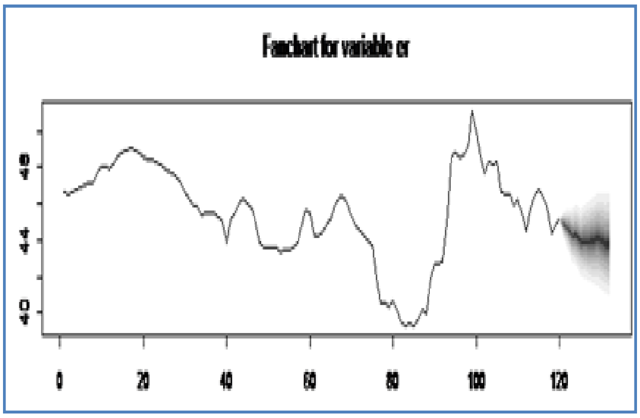

Fig. 10- Fan chart forecast of predicted exchange rate 2011

Neural Network Model using NeuralWorks Predict

Steps of applying the model are

- Loading data into Excel.
- Specifying the problem type.
- Specifying data and network characteristics.
- Saving the model.
- Training the model.
- Testing the model.
- Analyzing the results.
- Running new data through your model.
- Compute ME, MAE, MSE, RMSE, MPE and MAPE.

The monthly data of year 2000 to 2010 is used by the model, so a total of 120 records are accessed, out of which 83 records are used by the training data and 37 records are used by test data. [Table-5] shows the following statistical results using Neural Network Model:

- **R-** The linear correlation between the target values and the corresponding predicted output values.
- **Net-R-** The linear correlation between the target values and the raw network output values (before they are transformed into the measurement units of the problem).
- **Avg. Abs.-** The average absolute error between predicted output values and the corresponding target values.
- **Max. Abs.-** The maximum absolute error between predicted output values and the corresponding target values.
- **RMS-** The root mean square error between predicted output values and the corresponding target values.
- **Accuracy-** The percentage of predicted output values that are within the specified tolerance (20%) of the corresponding target values.
- **Conf. Interval-** The confidence interval is the range [target value ± confidence interval] within which the corresponding predicted output occurs 95% of the time.

Table 5- Records accessed by training and test data with 95% confidence interval

Exchange Rate	All	Train	Test
R	0.9479703	0.9539354	0.9352336
Net - R	0.92197	0.930453	0.774762
Avg. Abs.	0.640782	0.581056	0.774762
Max. Abs.	2.310219	2.310219	1.991043
RMS	0.855175	0.803548	0.960948
Accuracy (20%)	1	1	1
CI (95%)	1.682802	1.589903	1.944051
Records	120	83	37

The monthwise actual exchange rate of 2011 is known to us. [Table-6] shows the actual and predicted exchange rate of 2011 monthwise using Neural Network Model using NeuralWorks Predict [Fig-11] is a NeuralWorks Predict model screenshot generated when the training data is complete. It shows that six inputs, five hidden units and one output unit is taken and the exchange rate of 2011 is predicted.

[Fig-12] showing comparison between actual exchange rate of 2011 (shown by blue line) and predicted exchange rate of 2011 (shown by red line) using NeuralWorks Predict of Neural Network Model. The x-axis of the graph shows time period in months and the y-axis of the graph shows exchange rate values.

Table 6- Compares the actual exchange rate of 2011 with predicted exchange rate of 2011

Time Period	Actual Exchange Rate	Predicted Exchange Rate
Jan-11	45.375	44.3848152
Feb-11	45.3795	46.2393532
Mar-11	44.9143	42.9655075
Apr-11	44.301	42.8179855
May-11	44.9024	42.7414207
Jun-11	44.8109	43.1215172
Jul-11	44.396	42.7760849
Aug-11	45.3135	42.7414207
Sep-11	47.6905	42.7414207
Oct-11	49.202	42.8179855
Nov-11	50.6785	46.5665283
Dec-11	52.3824	47.0436554

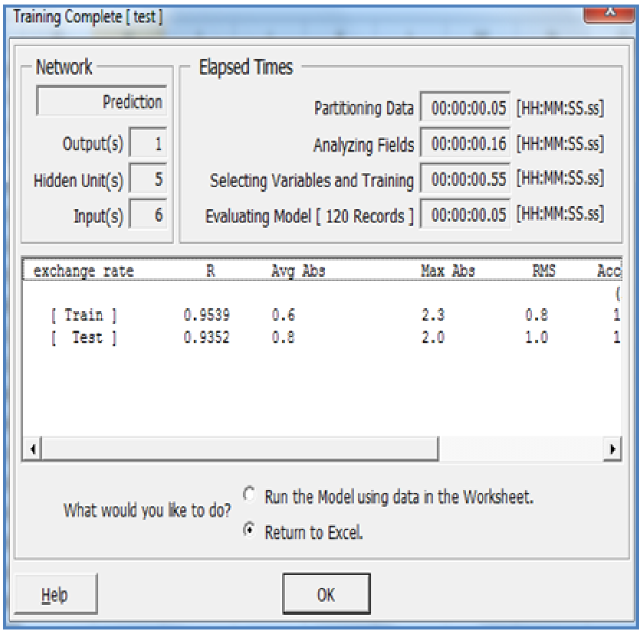

Fig. 11- shows that in the Neural Model 6 inputs are taken, with 5 hidden units and one output unit exchange rate is predicted

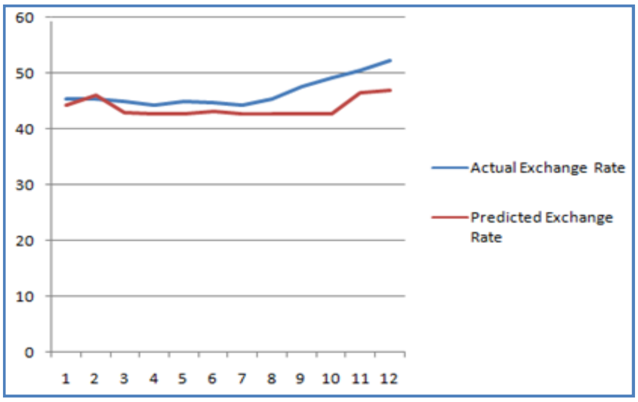

Fig. 12- Graph showing comparison between actual exchange rate of 2011 with predicted exchange rate of 2011 using NeuralWorks Predict

Performance Comparison

After fitting a time series model, one can evaluate it with forecast fit measures. When more than one forecasting technique seems reasonable for a particular application, then the forecast accuracy measures can also be used to discriminate between competing models. One can subtract the forecast value from the observed value of the data at that time point and obtain a measure of error. To evaluate the amount of this forecast error, the researcher may employ the mean error or the mean absolute error. Let A_t is the actual data and F_t represents the forecast and n denotes the number of forecasts made. Then the mean error (ME) is merely the average error.

$$ME = (1/n)\sum_{t=1}^{n} |(A_t - F_t)|$$

The mean absolute error (MAE) or mean absolute deviation (MAD) is calculated by taking the absolute value of the difference between the estimated forecast and the actual value at the same time so that the negative values do not cancel the positive values. The average of these absolute values is taken to obtain the mean absolute error.

$$MAE = (1/n)\sum_{t=1}^{n} |(A_t - F_t)|$$

The mean squared error (MSE) measure the variability in forecast errors. Obviously, the variability in forecast errors must be small. ME, MAE and MSE are all scale dependent measures of forecast accuracy, that is, their values are expressed in terms of the original units of measurement.

$$MSE = (1/n)\sum_{t=1}^{n} (A_t - F_t)^2$$

The root mean square error (RMSE) measures the average magnitude of the error. In it, the difference between forecast and corresponding observed values are each squared and then averaged over the sample. Finally, the square root of the average is taken. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. This means the RMSE is most useful when large errors are particularly undesirable.

$$RMSE = SQRT(MSE)$$

The percentage error (MPE) is the proportion of error at a particular point of time in the series. The average percentage error in the entire series is a general measure of fit useful in comparing the fits of different models. This measure adds up all of the percentage errors at each time point and divides them by the number of time points.

$$MPE = (100\%/n)\sum_{t=1}^{n} \frac{(A_t - F_t)}{A_t}$$

Because the positive and negative errors may tend to cancel themselves, MPE statistic is often replaced by the mean absolute percentage error (MAPE).

$$MAPE = (100\%/n)\sum_{t=1}^{n} \left|\frac{(A_t - F_t)}{A_t}\right|$$

MAPE is the more objective statistic indicator because the measure is in relative percentage and will not be affected by the unit of the forecasting series. The closer MAPE approaches zero, the better the forecasting results

The prediction errors are calculated using all mentioned performance measures i.e. ME, MAE, MSE, RMSE, MPE and MAPE for all the four models namely Multiple Linear Regression in Excel, Multiple Linear Regression of Dedicated Time Series Analysis in Weka, Vector Autoregressive Model in R and Neural Network Model using NeuralWorks Predict and shown in [Table-7].

Table 7- Performance of all Models are compared through ME, MAE, MSE, RMSE, MPE and MAPE

MODELS	ME	MAE	MSE	RMSE	MPE	MAPE
Multiple Linear Regression in Excel	-1.25	3.38	13.63	3.69	-3.08%	7.27%
Multiple Linear Regression of Dedicated Time Series Analysis in Weka	-1.09	1.56	3.58	1.89	-2.49%	3.41%
Vector Autoregressive Model in R	2.48	2.48	13.89	3.72	5.04%	5.04%
Neural Network Model using Neural-Works Predict	2.69	2.84	11.26	3.35	5.62%	5.94%

Which Performance Measure Is Best?

Following observations are made from the performance measures:

- It is usually best to report the root mean squared error (RMSE) rather than mean squared error (MSE), because the RMSE is measured in the same units as the data, rather than in squared units, and is representative of the size of a "typical" error.
- The mean absolute error (MAE) is also measured in the same units as the original data, and is usually similar in magnitude to, but slightly smaller than, the root mean squared error. The mathematically challenged usually find this an easier statistic to understand than the RMSE.
- The mean absolute percentage error (MAPE) is also often useful for purposes of reporting, because it is expressed in generic percentage terms which will make some kind of sense even to someone who has no idea what constitutes a "big" error in terms of dollars spent or widgets sold. The MAPE can only be computed with respect to data that are guaranteed to be strictly positive.
- The mean error (ME) and mean percentage error (MPE) that are reported in some statistical procedures are signed measures of error which indicate whether the forecasts are biased--i.e., whether they tend to be disproportionately positive or negative. Bias is normally considered a bad thing, but it is not the bottom line. Bias is one component of the mean squared error--in fact mean squared error equals the variance of the errors plus the square of the mean error. That is: MSE = VAR (E) + (ME) ^ 2.
- The root mean squared error is more sensitive than other measures to the occasional large error: the squaring process gives disproportionate weight to very large errors. If an occasional large error is not a problem in decision situation, then the MAE or MAPE may be a more relevant criterion. In many cases these statistics will vary in unison--the model that is best on one of them will also be better on the others--but this may not be the case when the error distribution has "outliers."
- If one model's RMSE is 30% lower than another's, that is probably very significant. If it is 10% lower, that is probably somewhat significant. If it is only 2% better, that is probably not significant. These distinctions are especially important when we are trading off model complexity against the error measures: it is probably not worth adding another independent variable to a regression model to decrease the RMSE by only a few more percent.
- The confidence intervals for one-step-ahead forecasts are based almost entirely on RMSE, the confidence intervals for the longer-horizon forecasts than can be produced by time-series models depend heavily on the underlying modeling assumptions, particularly assumptions about the variability of the trend. The confidence intervals for some models widen relative-

ly slowly as the forecast horizon is lengthened (e.g., simple exponential smoothing models with small values of "alpha", simple moving averages, seasonal random walk models, and linear trend models). The rate at which the confidence intervals widen is not a reliable guide to model quality: what is important is the model should be making the correct assumptions about how uncertain the future is. It is very important that the model should pass the various residual diagnostic tests and "eyeball" tests in order for the confidence intervals for longer-horizon forecasts to be taken seriously.

So the observation states that mostly the performance of compared models depends on RMSE, but sometimes MAE or MAPE. Also, If two models are generally similar in terms of their error statistics and other diagnostics, you should prefer the one that is simpler and/or easier to understand. The simpler model is likely to be closer to the truth, and it will usually be more easily accepted by others. Forecasts from the four models are compared together with the actual usage. It is observed that forecasts from the Dedicated Time Series Model using Weka are closer to the actual demands than the forecasts from other three models.

Conclusion

In nowadays technology, financial institutions are able to produce huge datasets that build a foundation for approaching these enormously complex and dynamic problems with data mining tools. With vast data and literature in this field the potential significant benefits of solving these problems motivated extensive research for years. The mostly used data mining application applied to the field of finance is the modeling (predicting or forecasting) of financial data. Almost every statistical and mathematical computational method has been explored and used for financial modeling.

In this paper, four models namely Multiple Linear Regression in Excel, Multiple Linear Regression of Dedicated Time Series Analysis in Weka, Vector Autoregressive Model in R and Neural Network Model using NeuralWorks Predict are analyzed using real time series data. The daily exchanges of currency between the U.S. and other nations have a major effect on international trade. There are many advantages and disadvantages associated with strong and weak U.S dollars. This study will be helpful to learners and experts alike as they choose the best approach to solving basic data-mining problems. This will help in reducing the lead time for getting the best prediction possible.

The models have shown favorable forecasting accuracy with Linear Regression of Dedicated Time Series Analysis in Weka outperforming the other three models.

References

[1] Hagan M.T. and Behr S.M. (1987) *IEEE Trans. Power Syst.*, PWRS-2(3).
[2] Moghram Rahman S. (1989) *IEEE Trans. Power Syst.*, 4, 1484 -1491.
[3] Taylor J.W. and McSharry P.E. (2007) *IEEE Trans. Power Syst.*, 22(4).
[4] Abu-El-Magd M.A. and Findlay R.D. (2003) *Electrical and Computer Engineering, IEEE CCECE Canada Conf.,* 3, 1723-1726.
[5] Perwej Y. and Perwej A. (2012) *International Journal of Computer Science, Engineering and Applications*, 2(2).
[6] Kosorus H., H¨onigl J. and K¨ung J. (2011) 22*nd International Workshop on Database and Expert Systems Applications*.
[7] Brockwell P.J. and Davis R.A (2006) *Time Series: Theory and Methods*, 2*nd ed.*, Springer, New York.
[8] Cowpertwait P.S.P. and Metcalfe A.V. (2009) *Introductory Time Series with R*, 1*st ed.*, Springer, New York.
[9] Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P. and Witten I.H. (2009) *SIGKDD Explorations*, 11(1).
[10] Shumway R.H. and Stoffer D.S. (2006) *Time Series Analysis and Its Applications - With R Examples*, 2nd ed., Springer, New York.
[11] Suhartono and Subanar, Guritno S. (2005) Jurnal Teknik Industri, 7(1), 22-30.
[12] Povinelli R.J. and Feng X. (2003) *IEEE Transactions on Knowledge and Data Engineering*, 15(2), 339-352.
[13] Alves da Silva A.P. and Moulin L.S. (2000) *IEEE Transactions on Power Systems*, 15(4), 1191-1196.
[14] Medeiros M.C., Veiga A. and Pedreira C.E. (2001) *IEEE Transactions on Neural Networks*, 12(4), 755-764.
[15] Hippert H.S., Pedreira C.E. and Souza R.C. (2001) *IEEE Trans. Power Syst.*, 16(1), 44-55.
[16] Cărbureanu M. (2011) *Economic Sciences Series*, LXIII(3), 105 -112.
[17] Weiss S.M. and Indurkhya N. (1999) *Predictive Data Mining: A Practical Guide*, 1*st ed*., Morgan Kauffmann Publishers.
[18] Han J. and Kamber M. (2006) *Data Mining: Concepts and Techniques* 2*nd ed*., Elsevier Inc.: Morgan Kaufmann Publishers.